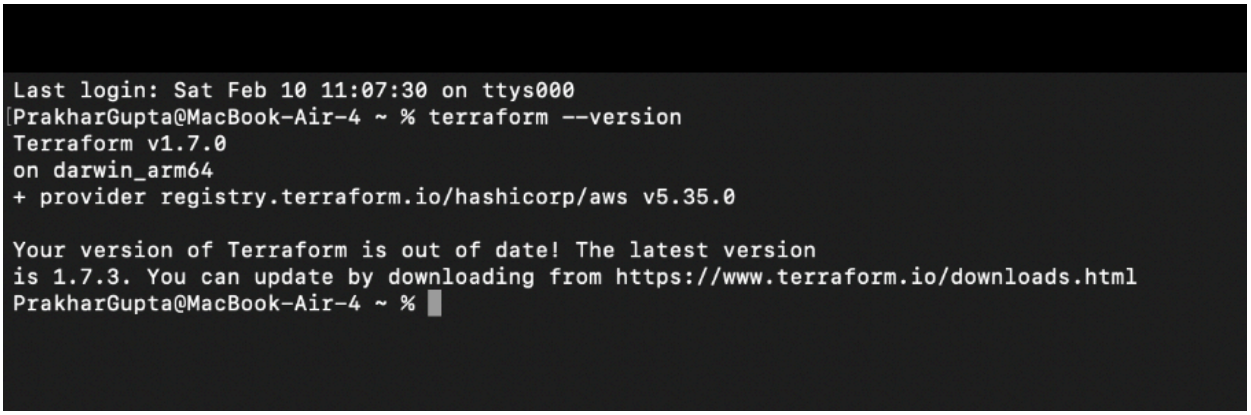
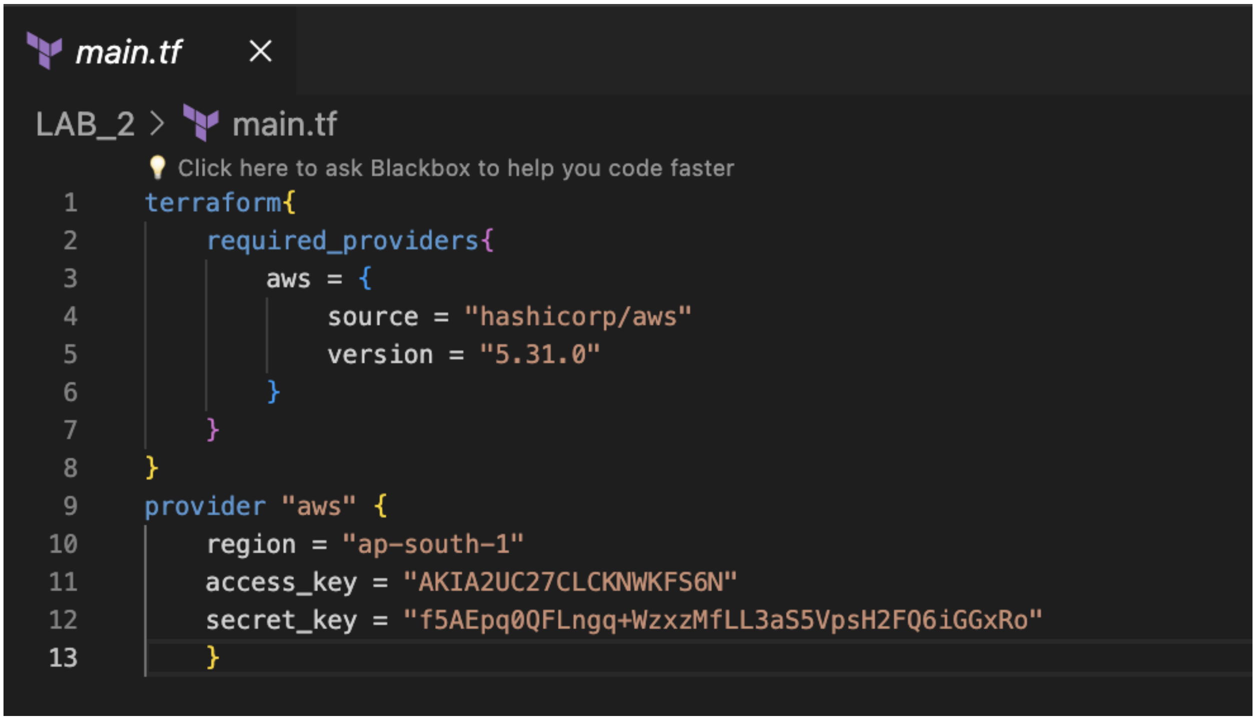
Name : Prakhar Gupta
SAP ID : 500093012

EXP 1 : Install Terraform In MacOS



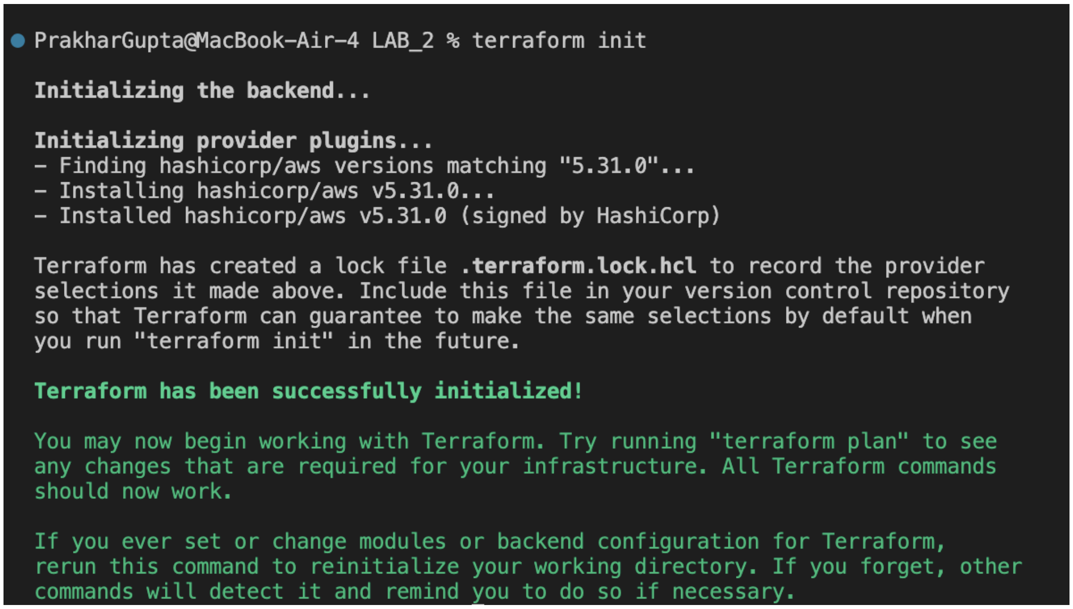
```
Last login: Sat Feb 10 11:07:30 on ttys000
[PrakharGupta@MacBook-Air-4 ~ % terraform --version
Terraform v1.7.0
on darwin_arm64
+ provider registry.terraform.io/hashicorp/aws v5.35.0

Your version of Terraform is out of date! The latest version
is 1.7.3. You can update by downloading from https://www.terraform.io/downloads.html
PrakharGupta@MacBook-Air-4 ~ %
```

EXP 2 : Terraform AWS provider IAM user setting



```
terraform{
    required_providers{
        aws = {
            source = "hashicorp/aws"
            version = "5.31.0"
        }
    }
}
provider "aws" {
    region = "ap-south-1"
    access_key = "AKIA2UC27CLCKNWKFS6N"
    secret_key = "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"
    }
```

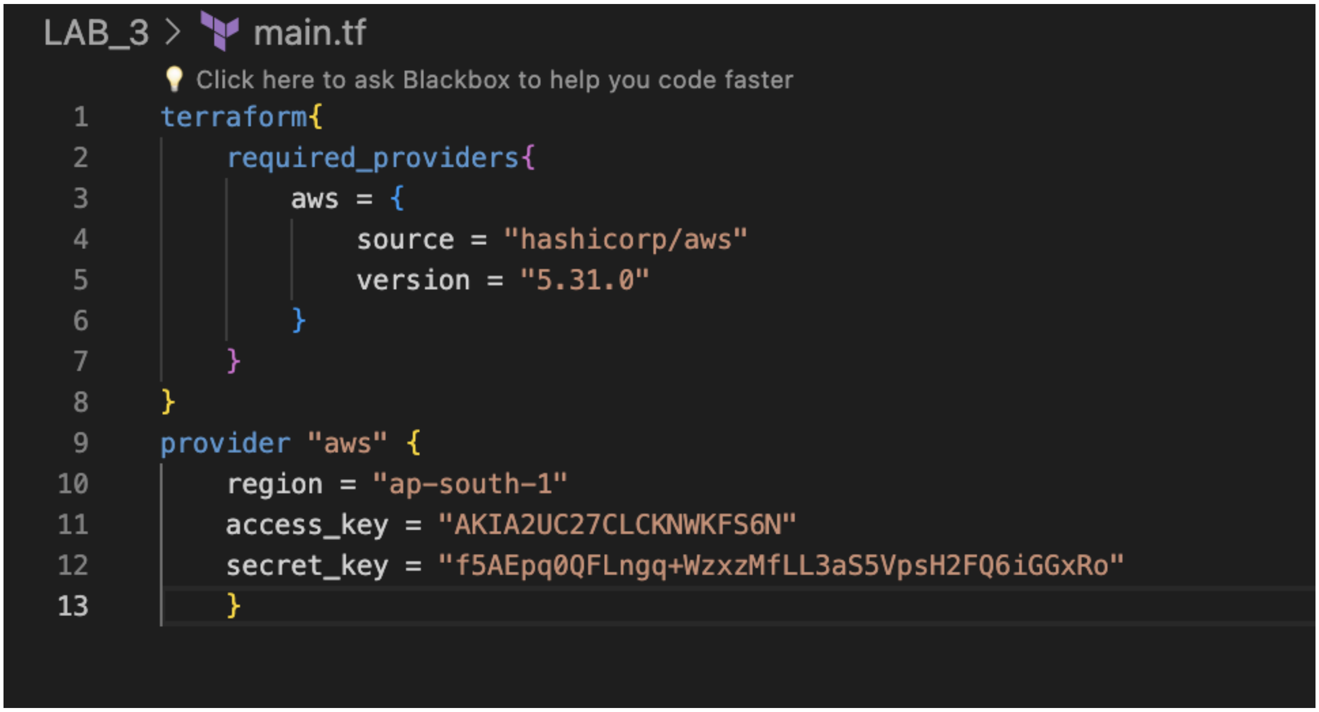
Terraform init



```
● PrakharGupta@MacBook-Air-4 LAB_2 % terraform init

Initializing the backend...

Initializing provider plugins...
- Finding hashicorp/aws versions matching "5.31.0"...
- Installing hashicorp/aws v5.31.0...
- Installed hashicorp/aws v5.31.0 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

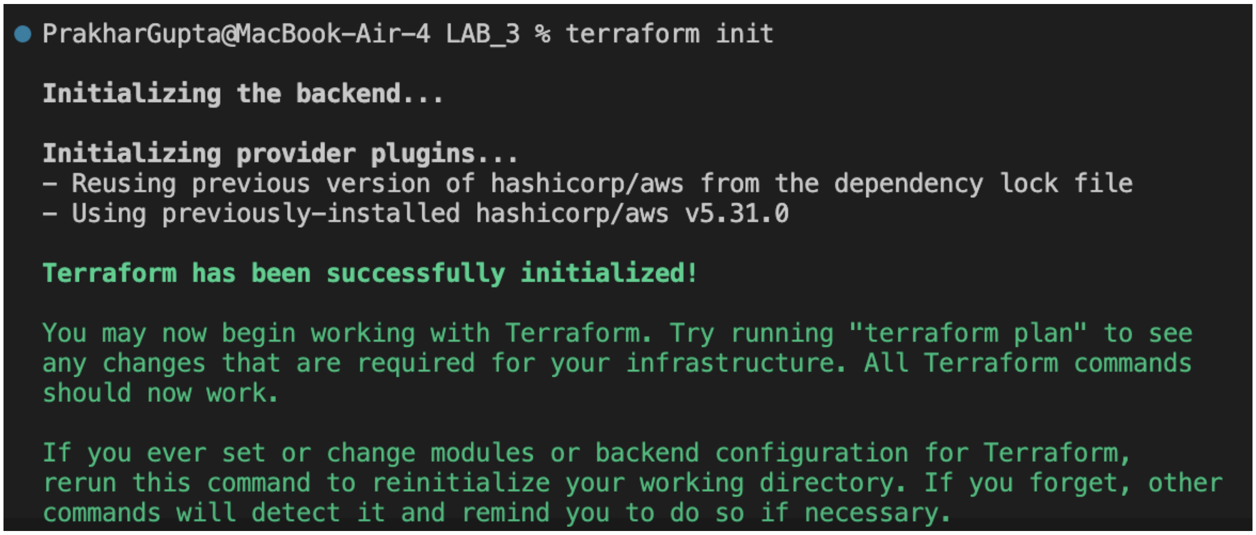
EXP 3 : Provisioning an EC2 Instance on AWS

Creating a file named main.tf



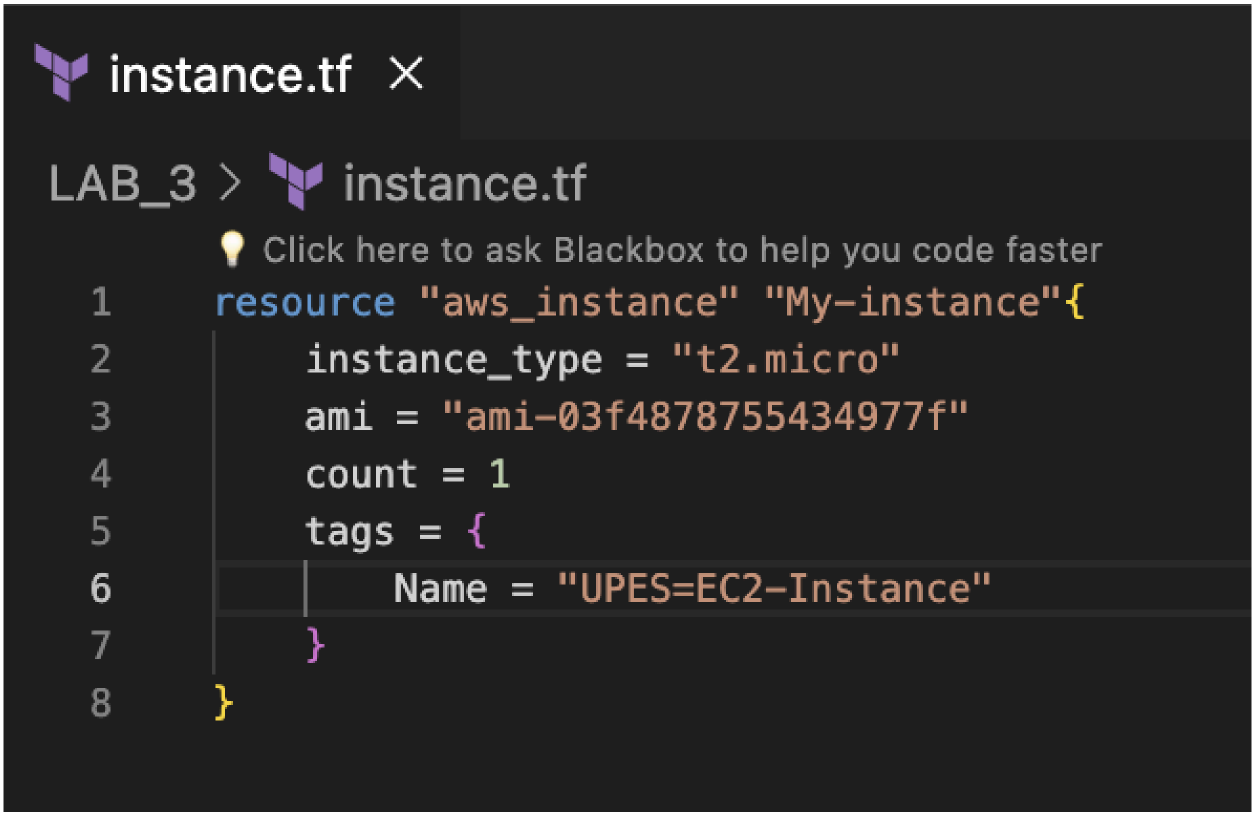
```
terraform{
    required_providers{
        aws = {
            source = "hashicorp/aws"
            version = "5.31.0"
        }
    }
}
provider "aws" {
    region = "ap-south-1"
    access_key = "AKIA2UC27CLCKNWKFS6N"
    secret_key = "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"
    }
```

Terraform init



```
PrakharGupta@MacBook-Air-4 LAB_3 % terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.31.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
```

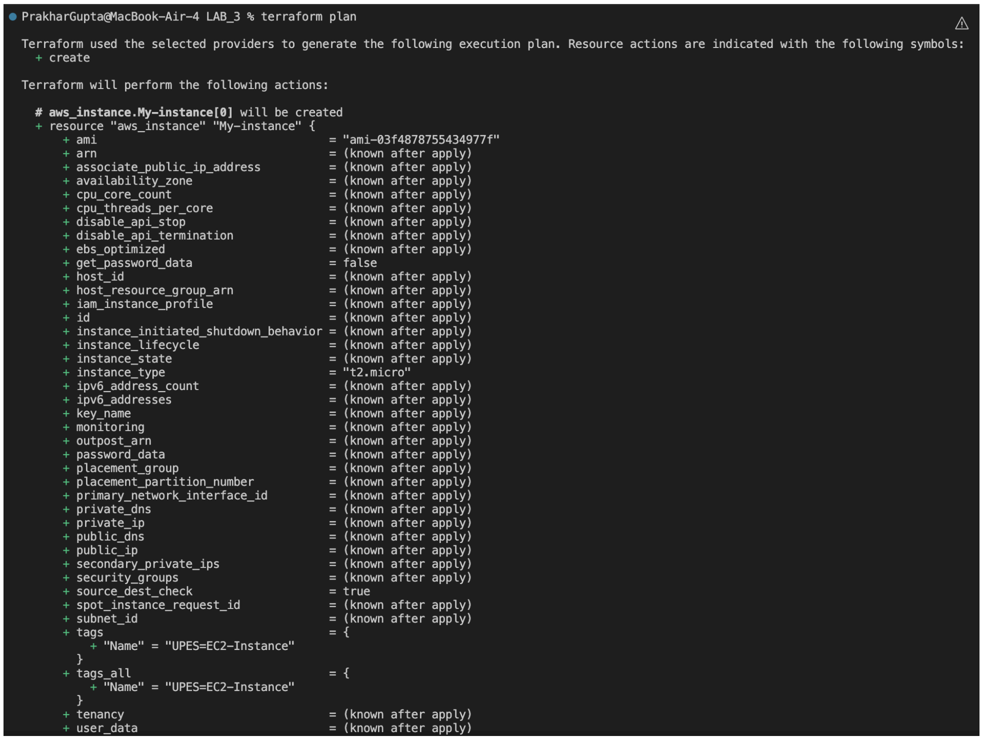
Creating a file named instance.tf



💡 Click here to ask Blackbox to help you code faster

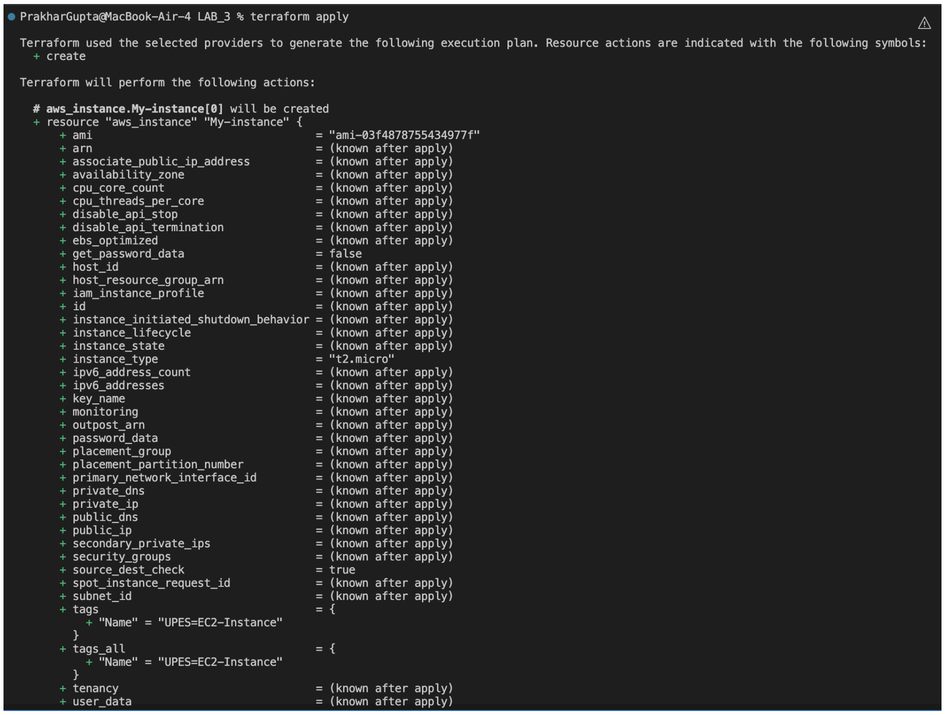
```
resource "aws_instance" "My-instance"{
    instance_type = "t2.micro"
    ami = "ami-03f4878755434977f"
    count = 1
    tags = {
        Name = "UPES=EC2-Instance"
    }
}
```

Terraform Plan

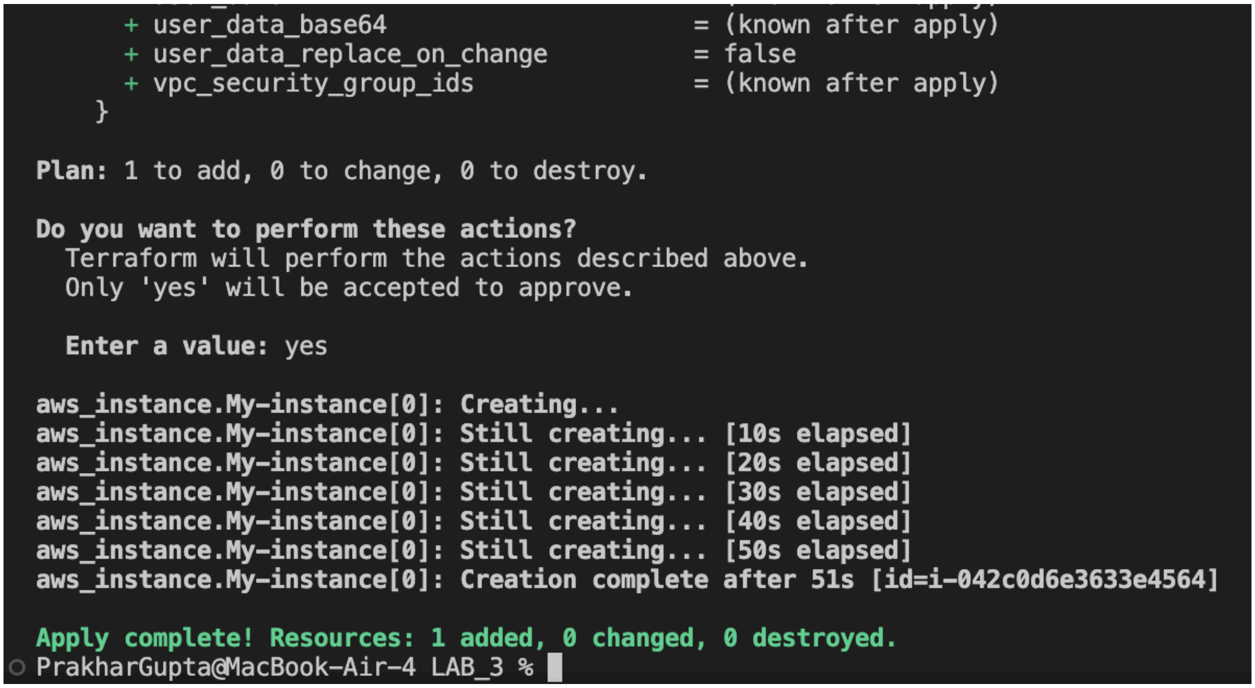


```
PrakharGupta@MacBook-Air-4 LAB_3 % terraform plan

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.My-instance[0] will be created
  + resource "aws_instance" "My-instance" {
      + ami                                  = "ami-03f4878755434977f"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "UPES=EC2-Instance"
        }
      + tags_all                             = {
          + "Name" = "UPES=EC2-Instance"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (known after apply)
```

Terraform Apply

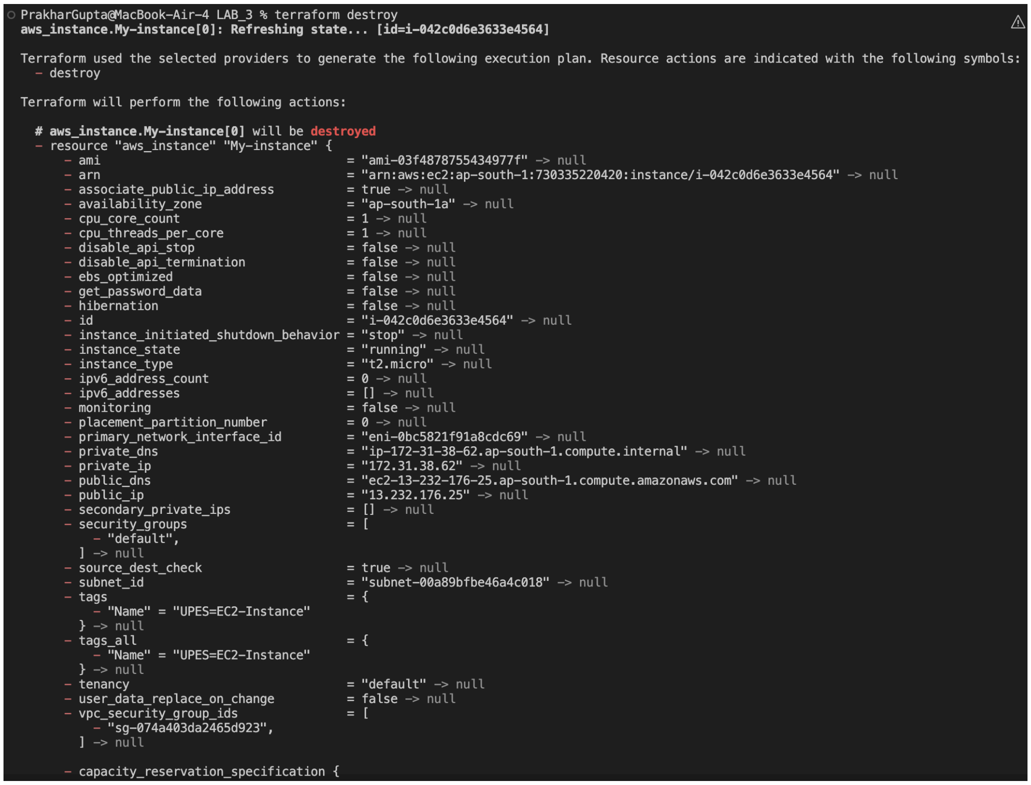


```
PrakharGupta@MacBook-Air-4 LAB_3 % terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
  + create

Terraform will perform the following actions:

  # aws_instance.My-instance[0] will be created
  + resource "aws_instance" "My-instance" {
      + ami                                  = "ami-03f4878755434977f"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = (known after apply)
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = (known after apply)
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t2.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = (known after apply)
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
      + placement_group                      = (known after apply)
      + placement_partition_number           = (known after apply)
      + primary_network_interface_id         = (known after apply)
      + private_dns                          = (known after apply)
      + private_ip                           = (known after apply)
      + public_dns                           = (known after apply)
      + public_ip                            = (known after apply)
      + secondary_private_ips                = (known after apply)
      + security_groups                      = (known after apply)
      + source_dest_check                    = true
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = (known after apply)
      + tags                                 = {
          + "Name" = "UPES=EC2-Instance"
        }
      + tags_all                             = {
          + "Name" = "UPES=EC2-Instance"
        }
      + tenancy                              = (known after apply)
      + user_data                            = (known after apply)
```

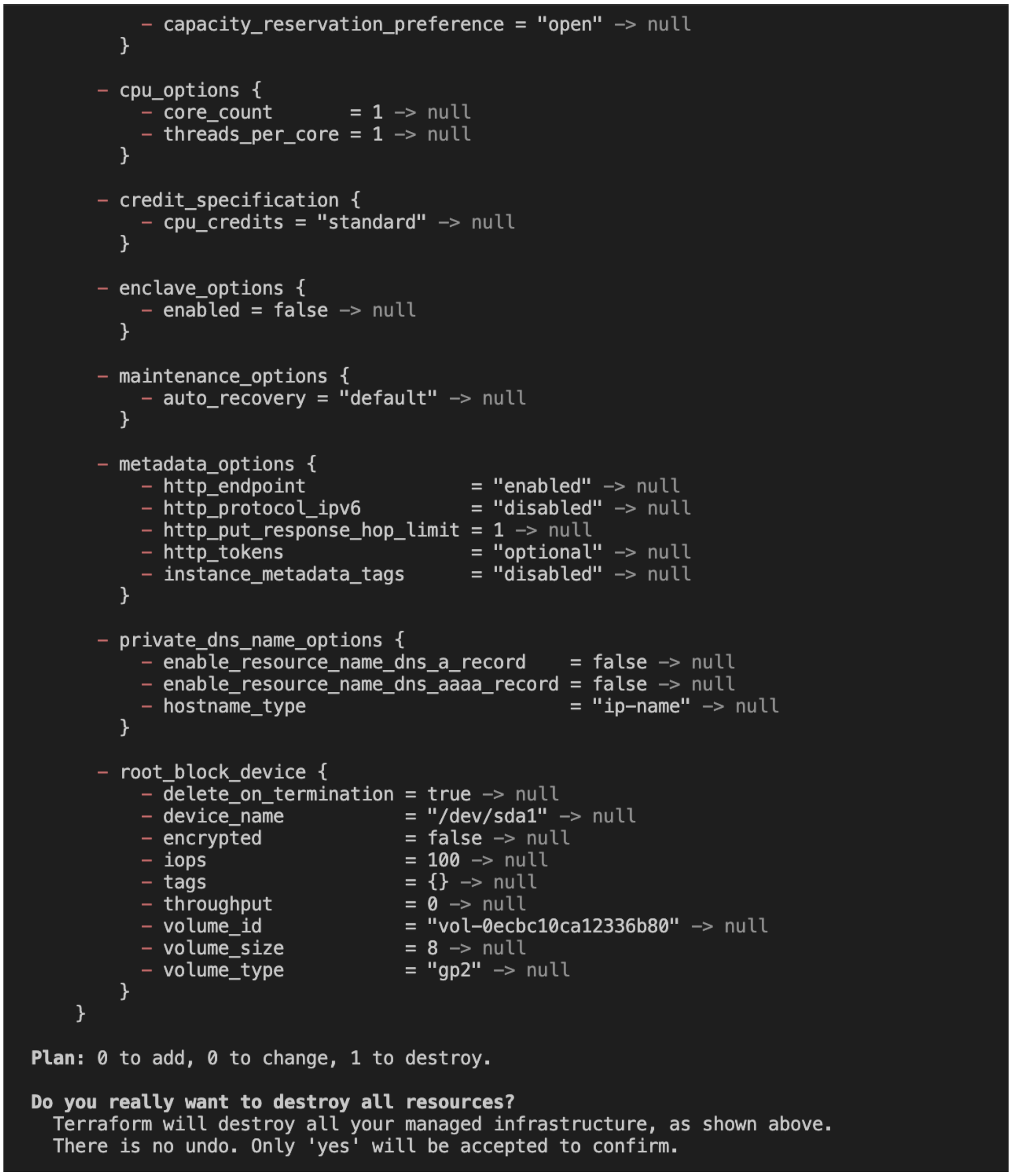


```
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = (known after apply)
    }

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.My-instance[0]: Creating...
aws_instance.My-instance[0]: Still creating... [10s elapsed]
aws_instance.My-instance[0]: Still creating... [20s elapsed]
aws_instance.My-instance[0]: Still creating... [30s elapsed]
aws_instance.My-instance[0]: Still creating... [40s elapsed]
aws_instance.My-instance[0]: Still creating... [50s elapsed]
aws_instance.My-instance[0]: Creation complete after 51s [id=i-042c0d6e3633e4564]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
○ PrakharGupta@MacBook-Air-4 LAB_3 %
```

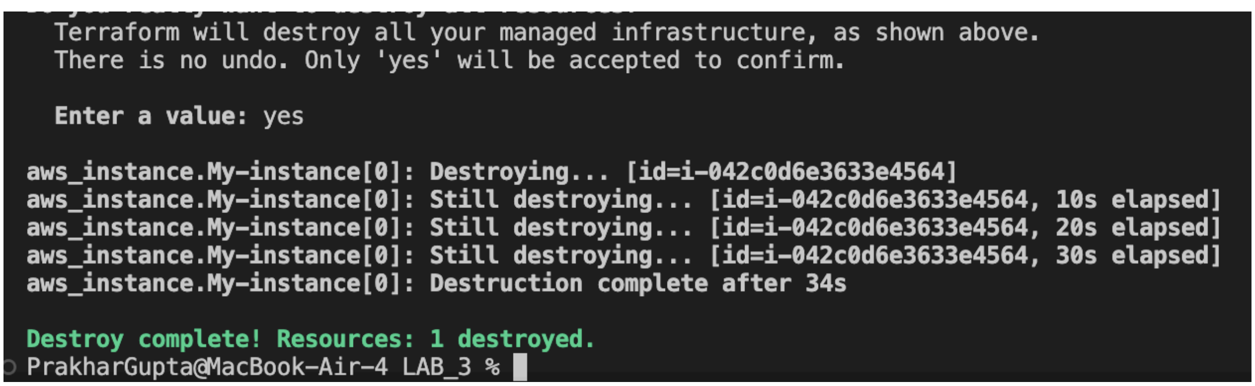
Terraform destroy



```
○ PrakharGupta@MacBook-Air-4 LAB_3 % terraform destroy
  aws_instance.My-instance[0]: Refreshing state... [id=i-042c0d6e3633e4564]

  Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:
    - destroy

  Terraform will perform the following actions:

    # aws_instance.My-instance[0] will be destroyed
    - resource "aws_instance" "My-instance" {
        - ami                                  = "ami-03f4878755434977f" -> null
        - arn                                  = "arn:aws:ec2:ap-south-1:730335220420:instance/i-042c0d6e3633e4564" -> null
        - associate_public_ip_address          = true -> null
        - availability_zone                    = "ap-south-1a" -> null
        - cpu_core_count                       = 1 -> null
        - cpu_threads_per_core                 = 1 -> null
        - disable_api_stop                     = false -> null
        - disable_api_termination              = false -> null
        - ebs_optimized                        = false -> null
        - get_password_data                    = false -> null
        - hibernation                          = false -> null
        - id                                   = "i-042c0d6e3633e4564" -> null
        - instance_initiated_shutdown_behavior = "stop" -> null
        - instance_state                       = "running" -> null
        - instance_type                        = "t2.micro" -> null
        - ipv6_address_count                   = 0 -> null
        - ipv6_addresses                       = [] -> null
        - monitoring                           = false -> null
        - placement_partition_number           = 0 -> null
        - primary_network_interface_id         = "eni-0bc5821f91a8cdc69" -> null
        - private_dns                          = "ip-172-31-38-62.ap-south-1.compute.internal" -> null
        - private_ip                           = "172.31.38.62" -> null
        - public_dns                           = "ec2-13-232-176-25.ap-south-1.compute.amazonaws.com" -> null
        - public_ip                            = "13.232.176.25" -> null
        - secondary_private_ips                = [] -> null
        - security_groups                      = [
            - "default",
          ] -> null
        - source_dest_check                    = true -> null
        - subnet_id                            = "subnet-00a89bfbe46a4c018" -> null
        - tags                                 = {
            - "Name" = "UPES=EC2-Instance"
          } -> null
        - tags_all                             = {
            - "Name" = "UPES=EC2-Instance"
          } -> null
        - tenancy                              = "default" -> null
        - user_data_replace_on_change          = false -> null
        - vpc_security_group_ids               = [
            - "sg-074a403da2465d923",
          ] -> null

        - capacity_reservation_specification {
```



```
          - capacity_reservation_preference = "open" -> null
        }

      - cpu_options {
          - core_count       = 1 -> null
          - threads_per_core = 1 -> null
        }

      - credit_specification {
          - cpu_credits = "standard" -> null
        }

      - enclave_options {
          - enabled = false -> null
        }

      - maintenance_options {
          - auto_recovery = "default" -> null
        }

      - metadata_options {
          - http_endpoint               = "enabled" -> null
          - http_protocol_ipv6          = "disabled" -> null
          - http_put_response_hop_limit = 1 -> null
          - http_tokens                 = "optional" -> null
          - instance_metadata_tags      = "disabled" -> null
        }

      - private_dns_name_options {
          - enable_resource_name_dns_a_record    = false -> null
          - enable_resource_name_dns_aaaa_record = false -> null
          - hostname_type                        = "ip-name" -> null
        }

      - root_block_device {
          - delete_on_termination = true -> null
          - device_name           = "/dev/sda1" -> null
          - encrypted             = false -> null
          - iops                  = 100 -> null
          - tags                  = {} -> null
          - throughput            = 0 -> null
          - volume_id             = "vol-0ecbc10ca12336b80" -> null
          - volume_size           = 8 -> null
          - volume_type           = "gp2" -> null
        }
    }

Plan: 0 to add, 0 to change, 1 to destroy.

Do you really want to destroy all resources?
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.
```



```
  Terraform will destroy all your managed infrastructure, as shown above.
  There is no undo. Only 'yes' will be accepted to confirm.

  Enter a value: yes

aws_instance.My-instance[0]: Destroying... [id=i-042c0d6e3633e4564]
aws_instance.My-instance[0]: Still destroying... [id=i-042c0d6e3633e4564, 10s elapsed]
aws_instance.My-instance[0]: Still destroying... [id=i-042c0d6e3633e4564, 20s elapsed]
aws_instance.My-instance[0]: Still destroying... [id=i-042c0d6e3633e4564, 30s elapsed]
aws_instance.My-instance[0]: Destruction complete after 34s

Destroy complete! Resources: 1 destroyed.
PrakharGupta@MacBook-Air-4 LAB_3 %
```